# PDR RID Report

**Originator** Tom Kalvelage **Phone No** 604-594-6556
**Organization** EDC DAAC
**E Mail Address** kalvelage@edcserver1.cr.usgs.gov
**Document** CSMS PDR Presentation

**RID ID PDR** 171
**Review** CSMS
**Originator Ref**
**Priority** 1

**Section** CSS Services **Page** NH-15 **Figure Table**

---

**Category Name** Design-CSS **Actionee** HAIS
**Sub Category** DOF and client-server
**Subject** Implications of DOF and client/server design

**Description of Problem or Suggestion:**

It is unclear from the presentation just how the DOF interacts with applications, and whether or not it tracks versions of the client and server halves of the application. There is some information on the nominal case of going to the central DOF database for object information, but not enough supporting information.

**Originator's Recommendation**

Describe what is gotten from the DOF, where replicated databases and caches are, when they are called, what happens if they are not there, what happens if there is a version mismatch, how to select a particular replicated database, how to update the replicants and caches, and if version control is integrated.

---

**GSFC Response by:** **GSFC Response Date**

**HAIS Response by:** Forman **HAIS Schedule** 2/10/95
**HAIS R. E.** Winston **HAIS Response Date** 2/10/95

It is unclear from the presentation just how the DOF interacts with applications,
(A) CSS is providing Encapsulation APIs through which application developers will gain access to the DOF services. RID PDR 143 refers to the need of a program development guide for application developers. This guide will detail how applications interact with the DOF as well as the rest of the CSS services. CSS will be providing a draft of this document by FOS CDR.

For a complete description of DOF, please see the design documentation provided in Section 6.3 of DID 305 CSMS Segment Design Document and Section 5.2 of DID 313 CSMS ICD document.

and whether or not it tracks versions of the client and server halves of the application.
(B) It does support version control. Each interface has a major and a minor version number. A modified interface with upward compatibility will have a new minor version number incremented. If the newly modified interface is not upward compatible, then a new major version number should be assigned. This is described in Section 3.2.1 of DID 313 CSMS ICD document.

There is some information on the nominal case of going to the central DOF database for object information, but not enough supporting information.
(C) The Distribution of the namespace is internal to the DCE COTs product. This is covered fully in the DCE documentation.

Describe what is gotten from the DOF,
(D) DOF is an integrated set of object services that provides the basic underlying communication framework for application developers to develop client/server applications in an object oriented network environment. DOF's basic (core) services include Directory/Naming, Security, Threads, Time and Remote Procedure Calls. Please see the design documentation for more details.

where replicated databases and caches are,
(E) Location of the actual namespace (master) and the replicas (slaves or copies) are configurable by the system administrator. Caches are kept near the client at each workstation. These are internal details of the DOF that the application developer doesn't have to worry about. As the very purpose of naming is to keep the location information transparently, the application developer should not have to be depending on where these databases/caches/services are physically kept.

when they are called, what happens if they are not there,
(F) Namespaces are not directly callable by the application. The application client specifies a service name, via the CSS service interface (API), to get the bindng information. It is the DOF which then gets that information from the namespace and returns it to the application client. The namespace has a master and at least one copy so that if one fails, applications still get information from the other. This also is transparent to the application developer.

what happens if there is a version mismatch

interface (API), to get the bindng information. It is the DOF which then gets that information from the namespace and returns it to the application client. The namespace has a master and at least one copy so that if one fails, applications still get information from the other. This also is transparent to the application developer.

what happens if there is a version mismatch,
(G) Namespaces don't have versions; The services (interfaces they provide) have version numbers, please refer to (B) above. If a compatible version cannot be found an error returned to the application (client) and the error is logged as an event (for use by MSS).

how to select a particular replicated database,
(H) Applications don't normally select a particular replica. The DOF returns information from an available replica or a master. Developers however, can specify whether the application is to get the information from the master only or from any replica (master or any available copy of the namespace).

how to update the replicants and caches, and if version control is integrated.
(I) Version control is associated with the interfaces of the services. The service information in stored in the namespace. There are two ways replicas can be updated: by the system administrator which updates the replicas immediately, and by an operation called "skulking", which periodically updates all the replicas from the master namespace.

---

**Status** **Closed** **Date Closed** **3/2/95** **Sponsor** **Broder**

****** **Attachment if any** ******

---